

ROLE BASED ACCESS CONTROL (RBAC)

John Barkley

RBAC Project Leader

Software Diagnostics and Conformance Testing

National Institute of Standards and Technology

(301) 975-3346

jbarkley@nist.gov

<http://hissa.nist.gov/rbac/>

ACTIVE PARTICIPANTS

- SDCT: Rick Kuhn, Bill Majurski,
Tony Cincotta, Alan Goldfine
- CSD: Dave Ferraiolo, Doctor Ramaswamy
Chandramouli
- GMU: Professor Ravi Sandhu, Jean Park
- UM: Doctor Virgil Gligor
- SETA: Ed Coyne, Ravi Sundaram (CRADA)
- VDG: Serban Gavrilla (contractor)

ROLE BASED ACCESS CONTROL (RBAC)

RBAC is an access control mechanism which:

- Describes complex access control policies.
- Reduces errors in administration.
- Reduces cost of administration.

NIST RBAC Activities

- NIST RBAC Model (Ferraiolo, Cugini, Kuhn)
- NIST RBAC Model Implementation for the WWW (RBAC/Web)
- Administrative tools: RBAC/Web Admin Tool & RGP-Admin
- Formal description of NIST RBAC Model in PVS (software specification in mathematical language)
- Test assertions and test software
- Cost model and role engineering tools
- Two patent applications and a provisional patent application

INDUSTRY RECOGNITION

- **IBM**'s patent application for IBM RBAC model cited NIST work as “closest prior art” (now implemented by **Tivoli**)
- **Sybase** and **Secure Computing** implemented NIST RBAC Model
- **Siemens Nixdorf** implemented parts of NIST RBAC Model in Trusted Web and references our work on their Web site
- NIST RBAC Model to be included in **Educom** IMS Specification
- Received **1998 Excellence in Technology Transfer Award** from **Federal Laboratory Consortium**

Page 15 of ITL Brochure

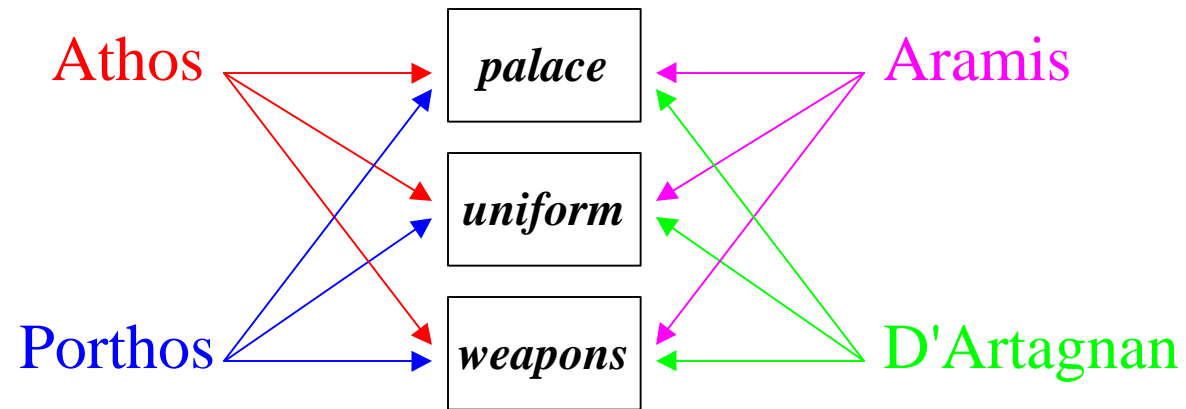
“I would like to take this opportunity to underscore the importance and relevance of research conducted by your laboratory into Role-Based Access Control (RBAC). In the area of security one of the features most requested by Sybase customers has been RBAC. They view this feature as indispensable for the effective management of large and dynamic user populations.”

Thomas J. Parenty
Director, Data and Communications Security
Sybase, Inc.
Emeryville, Ca.

RBAC MECHANISM

- Users are associated with roles.
- Roles are associated with permissions.
- A user has a permission only if the user has an authorized role which is associated with that permission.

Example: The Three Musketeers (User/Permission Association)



Example: The Three Musketeers (RBAC)

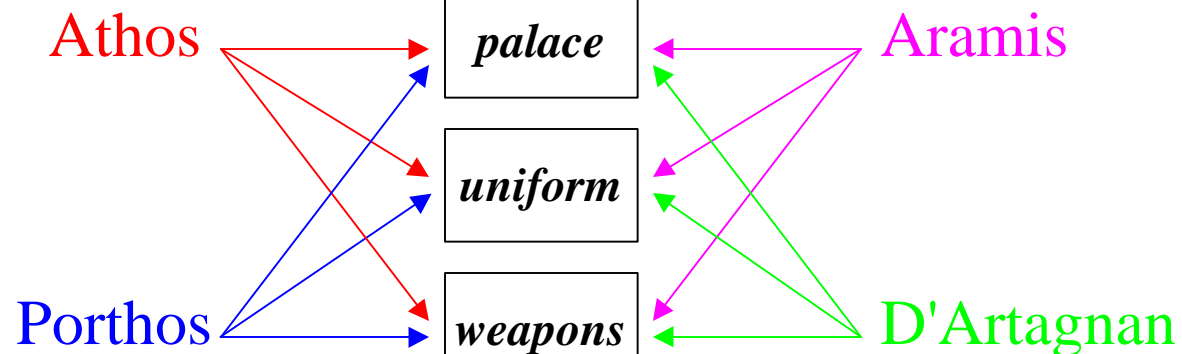
Athos
Porthos
Aramis
D'Artagnan

Musketeer

palace

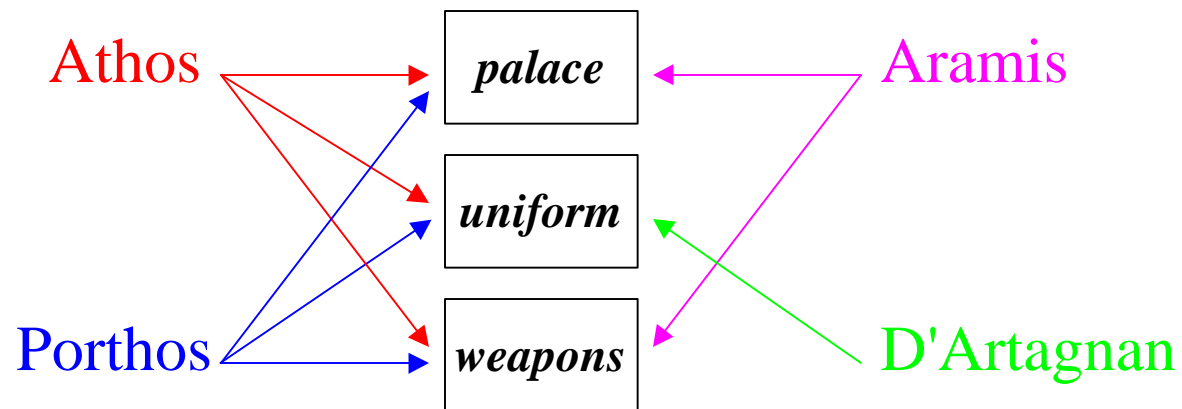
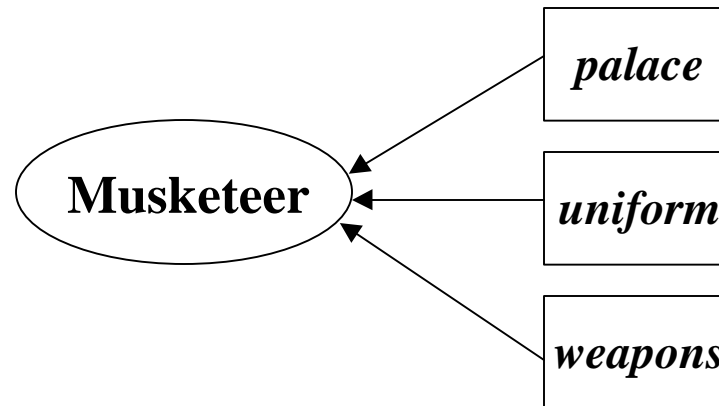
uniform

weapons

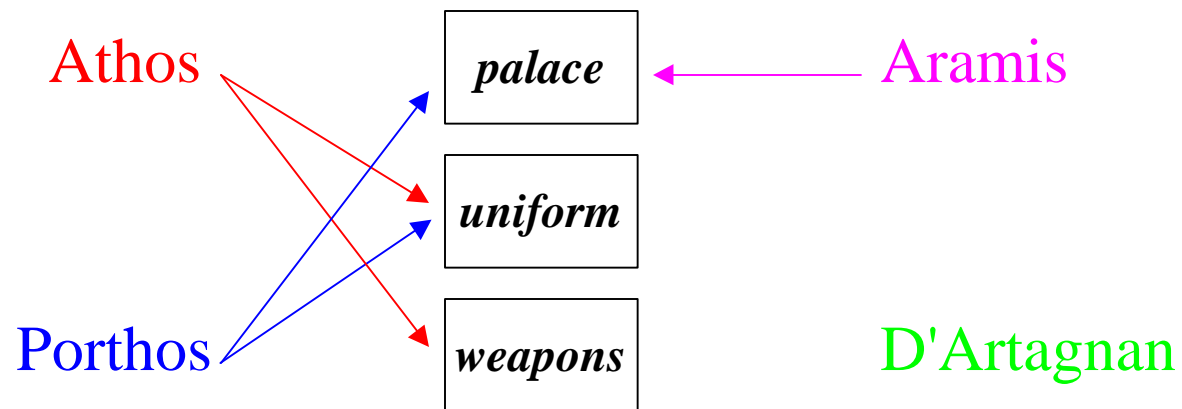
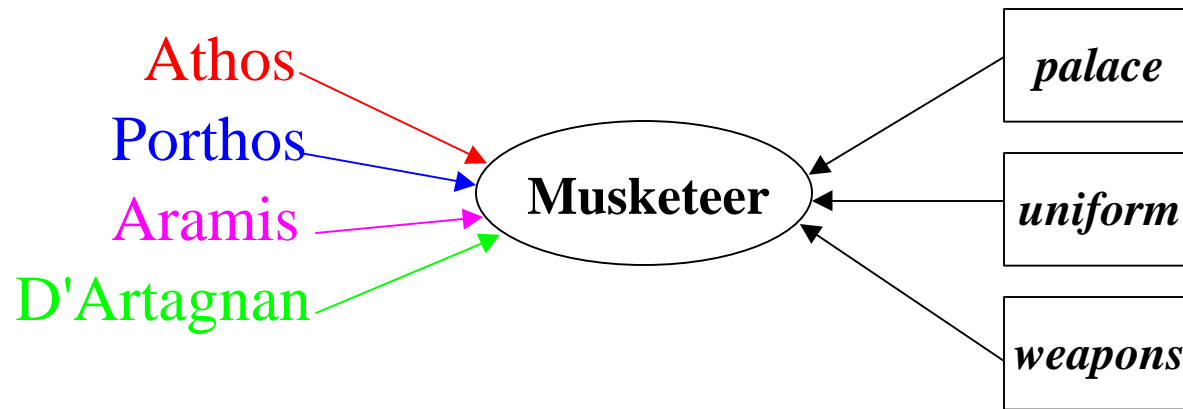


Example: The Three Musketeers (RBAC)

Athos
Porthos
Aramis
D'Artagnan



Example: The Three Musketeers (RBAC)



Quantifying RBAC Advantage

- For each job position, let:

U = Number of individuals in job position

P = Number of permissions required for job position

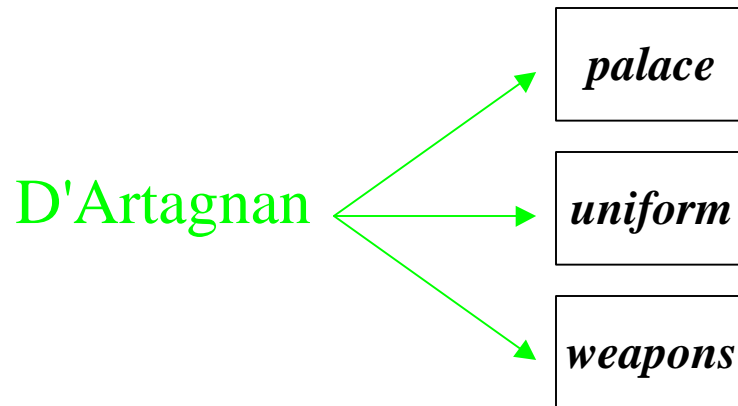
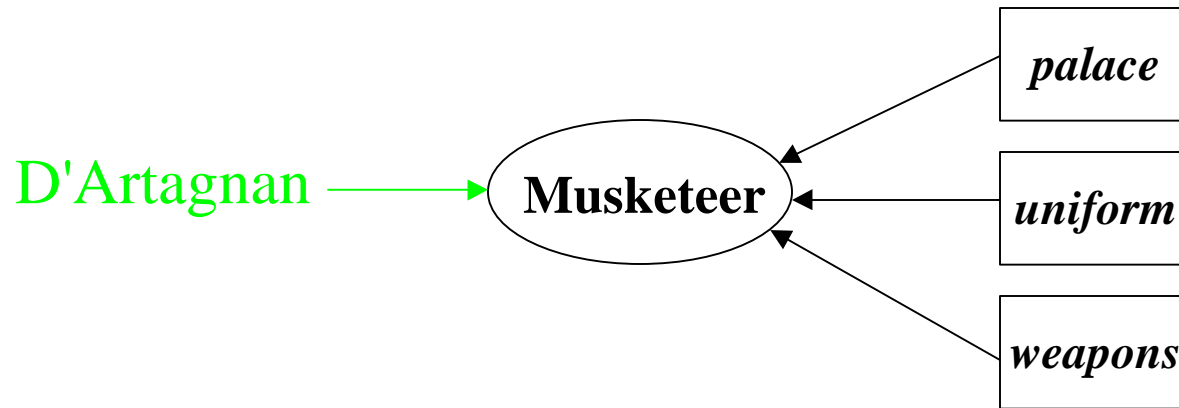
$$(U + P) < (U \cdot P) \Rightarrow \text{RBAC advantage}$$

$$U, P > 2 \Rightarrow (U + P) < (U \cdot P)$$

- For all job positions,

$$\sum_i^{n_{jp}} (U_i + P_i) < \sum_i^{n_{jp}} (U_i \cdot P_i) \Rightarrow \text{RBAC advantage}$$

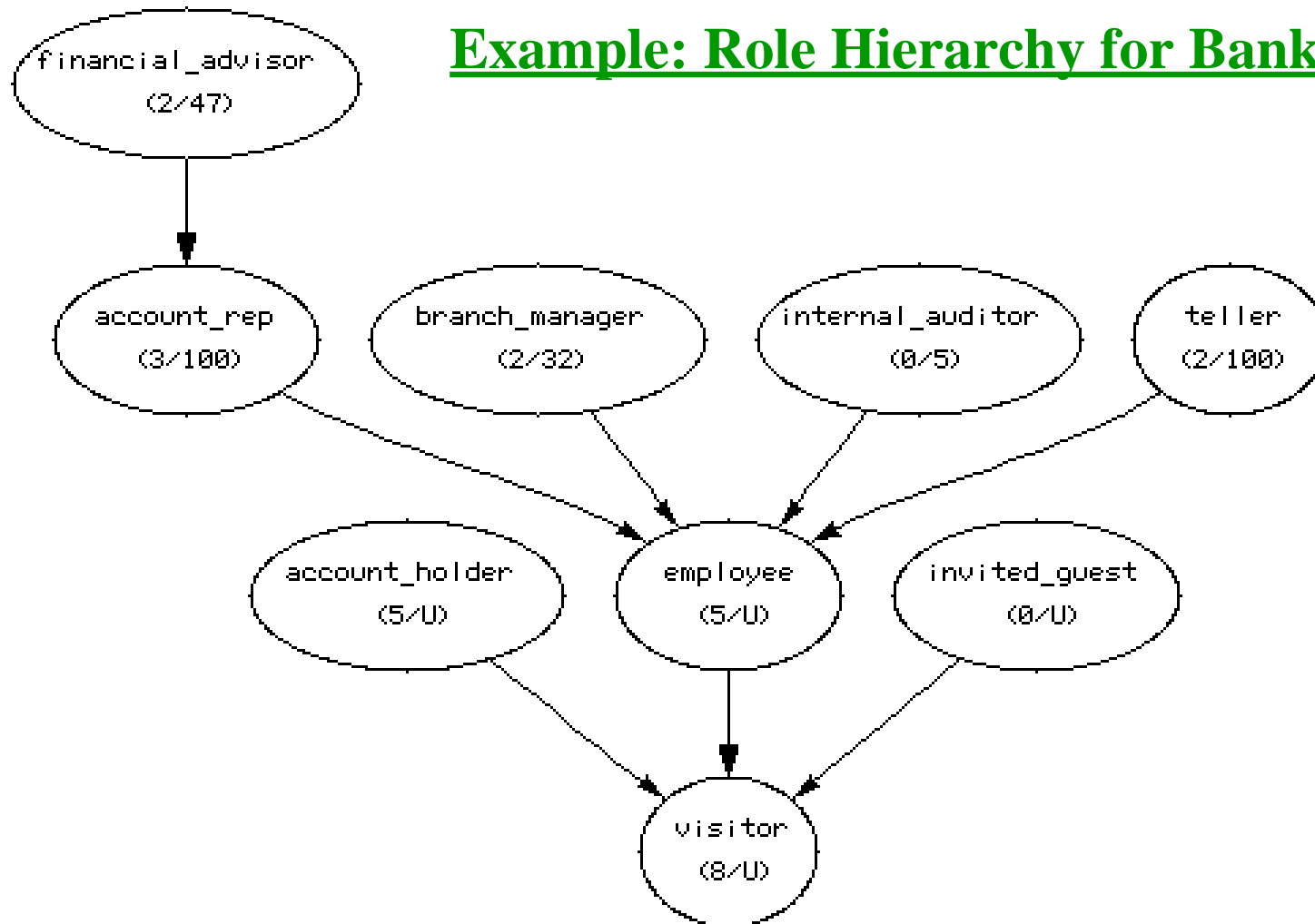
Example: (D'Artagnon becomes a Musketeer)



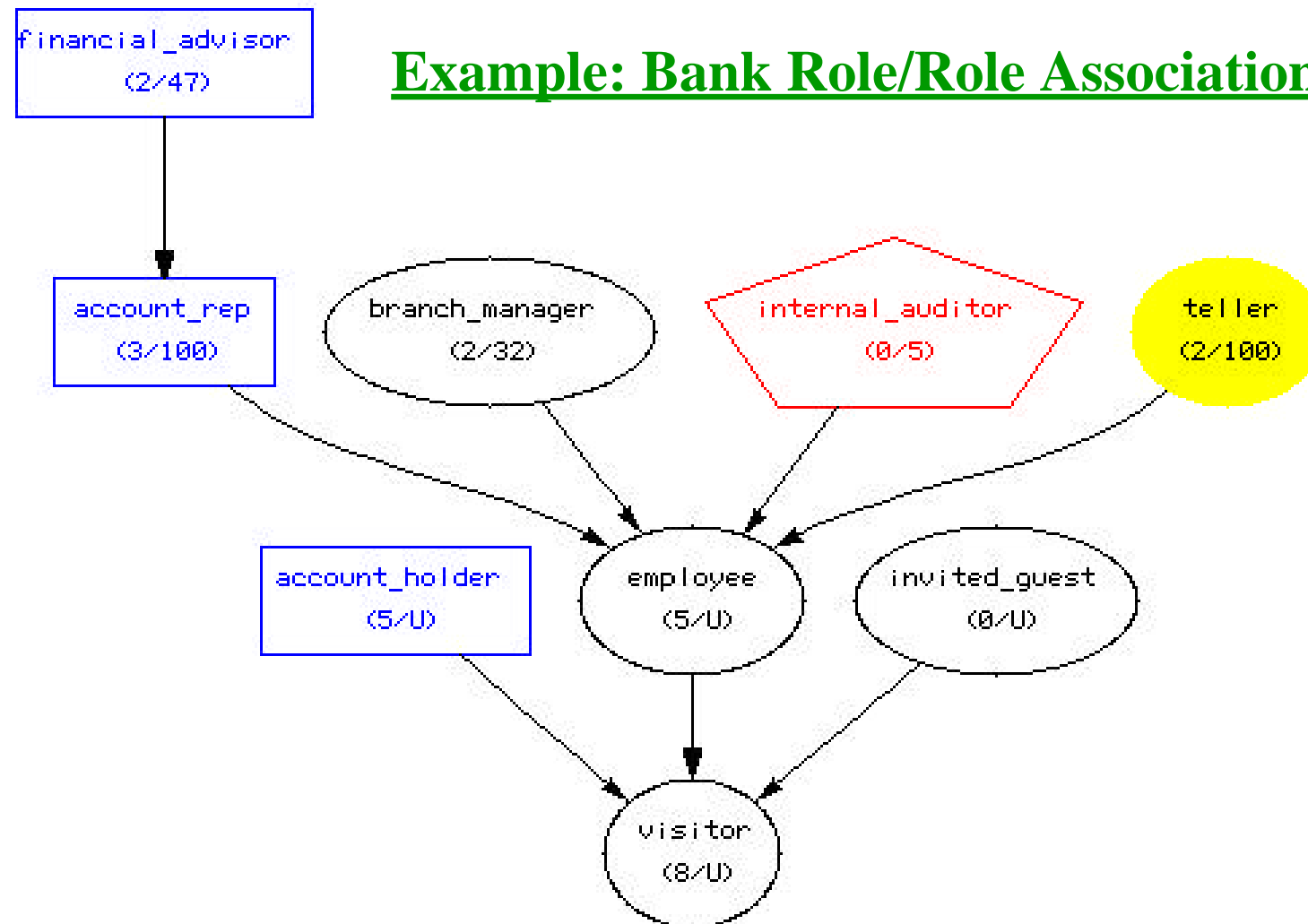
NIST RBAC Model

- Role Hierarchies, e.g, teller inherits employee
- Conflict of Interest Constraints:
 - Static Separation of Duty: user cannot be authorized for both roles, e.g., teller and auditor
 - Dynamic Separation of Duty: user cannot act simultaneously in both roles, e.g., teller and account holder
- Role Cardinality: maximum number of users authorized for role, e.g., branch manager

Example: Role Hierarchy for Bank



Example: Bank Role/Role Associations



RBAC Administrative Tools

- RBAC Admin Tool: user/role and role/role associations (RBAC/Web, NT, RDBMS)
- RGP-Admin: role/permission associations (NT)
- AccessMgr: Manipulation of all features of Windows NT ACLs
- Tool building with visual components
- Role Engineering and Diagnostic Tool

RBAC/Web Admin Tool: Main Display

Administrator Menu - Netscape
File Edit View Go Communicator Help
Go to: http://hissa.ncsl.nist.gov/rbac/

Display User:
Delete User:

User Selected: ko

Assigned roles
account_holder
teller

< Assign <
> De-assign >

Assignable roles
account_rep
branch_manager
financial_advisor
invited_guest

Not assignable roles
employee: already authorized
internal_auditor: in SSD with assigned role teller
role_admin: cardinality at maximum 1
visitor: already authorized

Role Administration

Add Role:

Display Role:
Delete Role:
Cardinality:

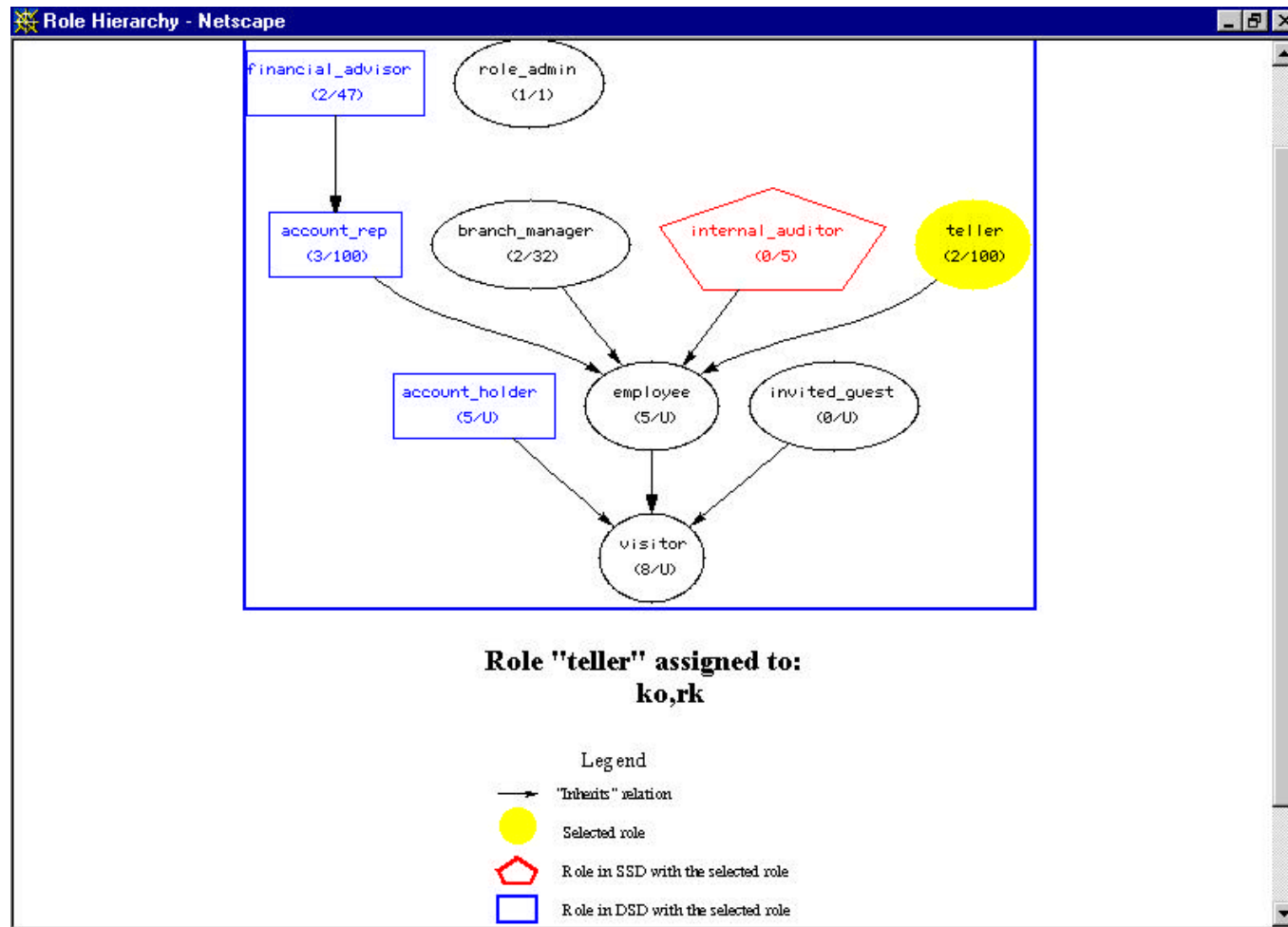
account_rep
branch_manager
employee
financial_advisor
internal_auditor
invited_guest
role_admin
teller
visitor

Role: teller
Cardinality: 100
Authorized users: 2

Inherit Disinherit
Set ssd Drop ssd
Set dsd Drop dsd

Document: Done

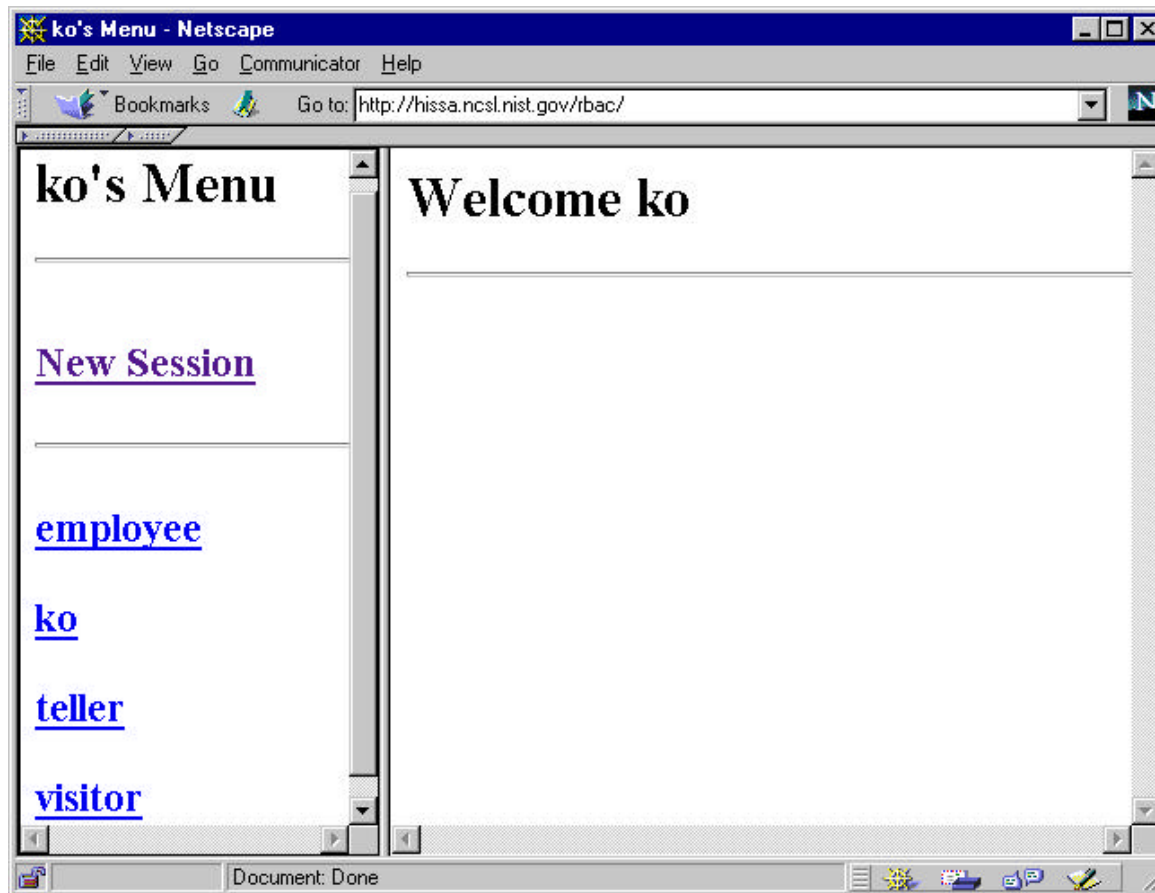
RBAC/Web Admin Tool: Graphical Display



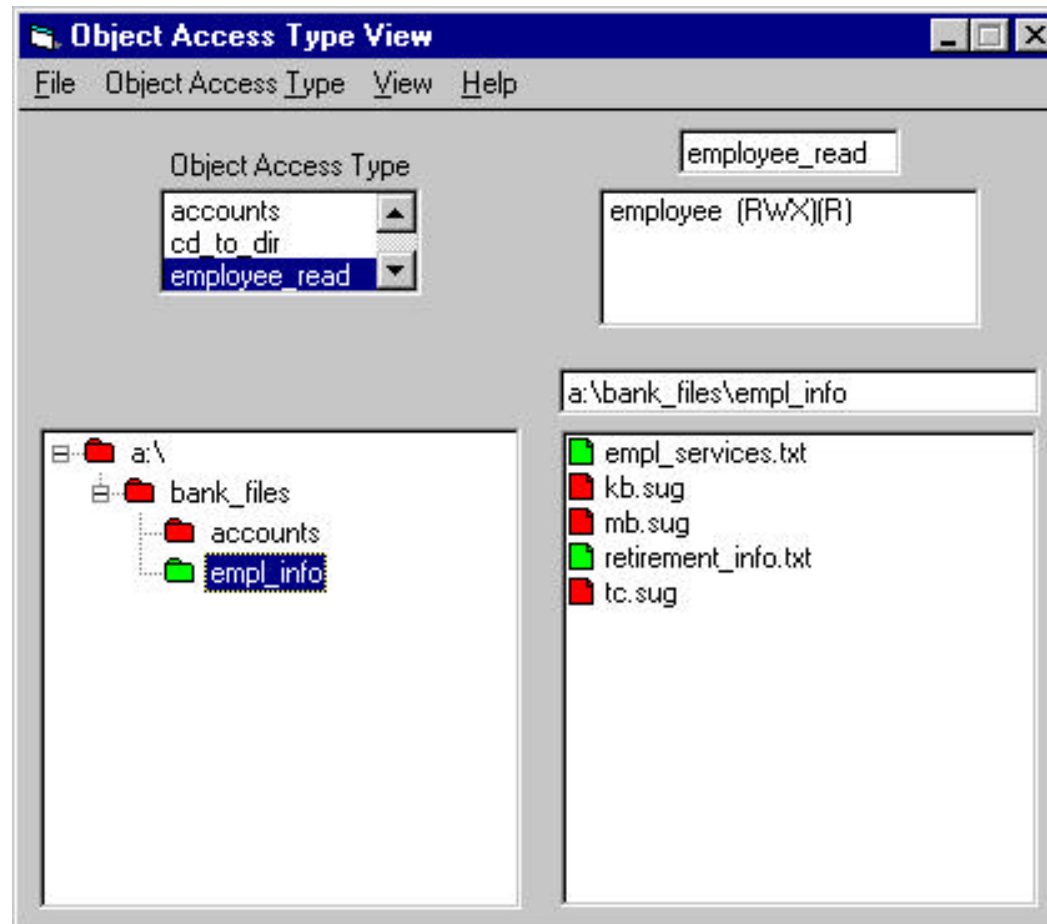
RBAC/Web login screen for ko



RBAC/Web login screen for ko



RGP-Admin: Object Access Type Window



RGP-Admin: Object Access Type Edit Window

Object Access Type Editor

Object Access Type Help

Object Access Type

- cd_to_dir
- employee_read
- suggestions**

suggestions

Role/Group

- account_rep
- financial_advisor
- teller

branch_manager ()(D)

employee (X)(R)

Directory Permissions

☐ R ☐ W ☐ X ☐ D

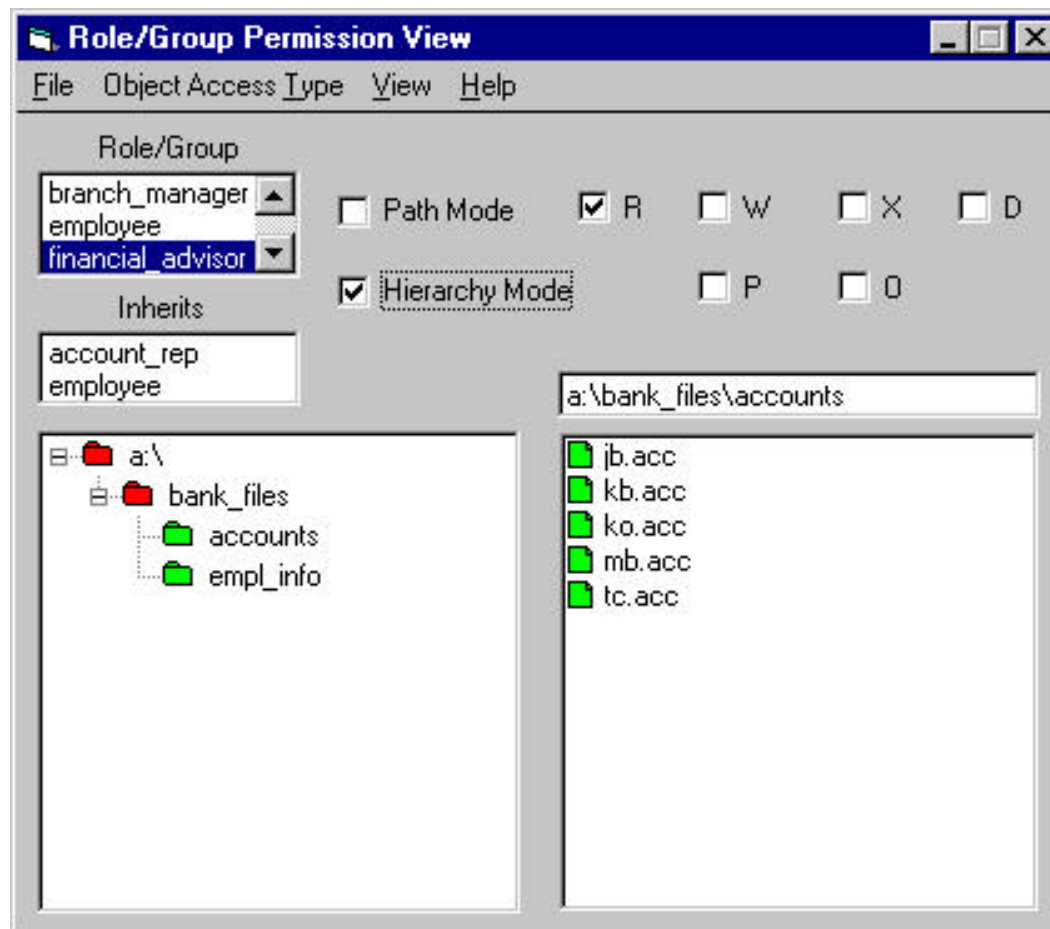
☐ P ☐ O

File Permissions

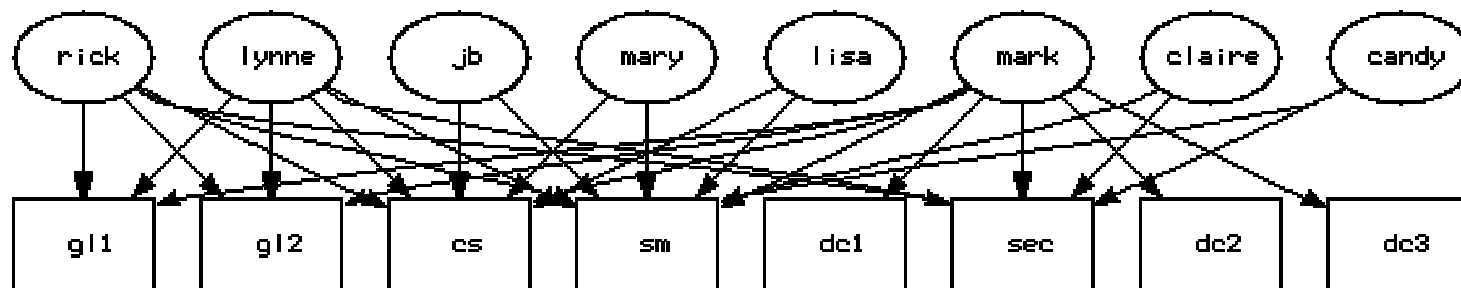
☐ R ☐ W ☐ X ☐ D

☐ P ☐ O

RGP-Admin: Role/Group Permission Window

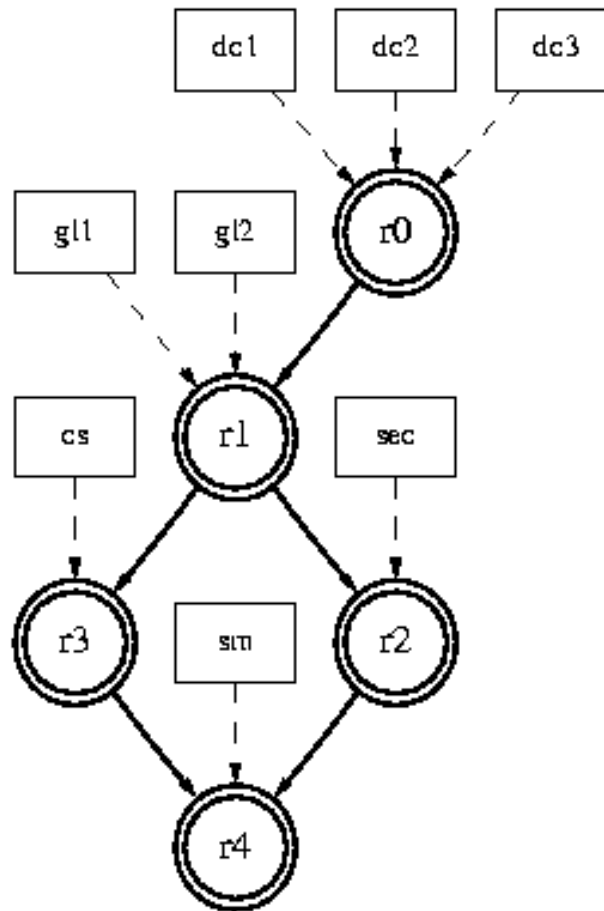


Role Engineering and Diagnostic Tool: input



Number of user/permission associations: 28

Role Engineering Tool: role/permission output



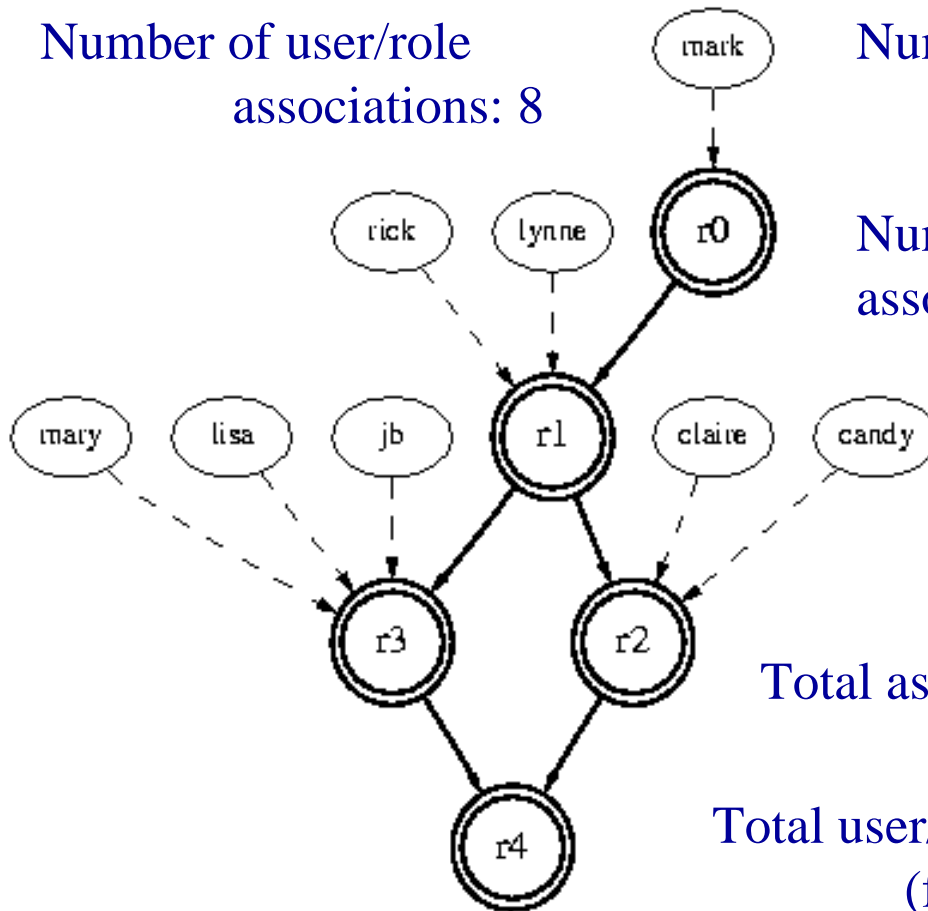
Number of role/permission associations: 8

Number of associations for role hierarchy: 5

Role Engineering Tool: user/role output

Number of user/role
associations: 8

Number of associations for
role hierarchy: 5



Number of role/permission
associations: 8 (previous slide)

Total associations with RBAC: 21

vs.

Total user/permission associations: 28
(from earlier slide)